

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-093577

(43)Date of publication of application : 04.04.1997

(51)Int.Cl.

H04N 7/24  
H04N 5/92  
H04N 7/08  
H04N 7/081  
// H03M 7/00

(21)Application number : 07-251253

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 28.09.1995

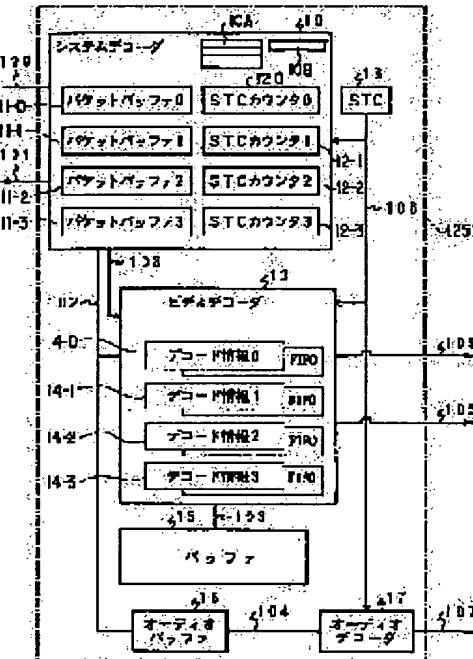
(72)Inventor : SHIBAHARA AKIHIKO

## (54) MOVING IMAGE DECODER

### (57)Abstract:

**PROBLEM TO BE SOLVED:** To process plural MPEG data streams with a single MPEG(moving picture image coding expert group) hardware decoder.

**SOLUTION:** A system decoder 10 processing an MPEG data stream is provided with plural packet buffers 11-0, 11-1,... and a function of controlling an output of data stored in the packet buffers synchronously with a system time clock supplied from an STC(system time clock) module 18. Then a video decoder 13 is provided with plural record information registers 14-0, 14-1,... storing plural sets of decode information corresponding to plural data streams respectively and plural data buffers (FIFO) storing data of decode object to process plural moving image data streams with a single hardware decoder.



**\* NOTICES \***

JPO and INPIT are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

---

**CLAIMS**

---

**[Claim(s)]**

[Claim 1]A system decoder which is provided with the following and characterized by enabling it to treat two or more video data streams with a single hardware resource.

Two or more packet buffers which store two or more video packets individually, respectively.

A means which carries out the output control of the data stored in two or more of these packet buffers synchronizing with a signal supplied from the exterior.

A means to notify a state of two or more above-mentioned packet buffers outside.

[Claim 2]The system decoder with a means to identify stream ID contained in a system stream, and to determine a storage location packet buffer according to claim 1.

[Claim 3]The system decoder with a means which carries out updating control of the synchronous data contained in a data stream corresponding to two or more packet buffers of each according to claim 1.

[Claim 4]The system decoder according to claim 1 which builds in two or more FIFO buffers which constitute two or more packet buffers, an STC counter for synchronous control provided corresponding to this FIFO buffer, and a control register which manages data of the above-mentioned FIFO buffer.

[Claim 5]Two or more registers which store individually two or more sets of decode information corresponding to each data stream for decoding two or more video data streams, respectively are built in, A video decoder characterized by enabling it to perform reproduction of two or more video data streams with a single hardware resource.

[Claim 6]A video decoder which is provided with the following and characterized by enabling it to perform reproduction of two or more video data streams with a single hardware resource.

Two or more registers which store individually two or more sets of decode information corresponding to each data stream for decoding two or more video data streams, respectively. A FIFO buffer which stores data which is provided corresponding to this register and serves as a candidate for decoding, respectively.

A decoder main part which decodes data stored in a FIFO buffer [ / based on decode information which chose one of the above-mentioned registers, and was stored in the register concerned ].

[Claim 7]A video decoding device which is provided with the following and characterized by two or more video data streams being renewable in real time with a single hardware resource.

A system decoder with two or more memory measures which store two or more video data streams individually, respectively.

A video decoder with two or more memory measures which store individually two or more sets of decode information for reproducing two or more video data streams stored in a memory measure of this system decoder, respectively.

[Claim 8]An MPEG decoder comprising:

A system decoder with two or more memory measures which store individually two or more

video data streams containing sound data, respectively.

A video decoder with two or more memory measures which store individually two or more sets of decode information for reproducing two or more video data streams stored in a memory measure of this system decoder, respectively.

An operating buffer with which decoding including frame data processing of this video decoder is presented.

A sound buffer and a sound decoder which carry out the reproducing output of the sound data in the above-mentioned data stream synchronizing with video.

---

[Translation done.]

**\* NOTICES \***

JPO and INPI are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.\*\*\*\* shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

---

**DETAILED DESCRIPTION**

---

**[Detailed Description of the Invention]****[0001]**

[Field of the Invention]This invention relates to the video decoding device which enabled realization of the regenerative function of two or more sorts of video data streams (for example, MPEG system stream) with the single hardware resource.

**[0002]**

[Description of the Prior Art]Real Magic made from SIGMA DESIGNS is one of typical things of the MPEG hardware decoder for reproducing the dynamic image data compressed by MPEG. The performance of this hardware decoder can decode the color video of 352x240 dot size at the rate of 30fps (frame per second).

[0003]However, in the above-mentioned conventional hardware decoder, a renewable MPEG data stream is restricted to one (1), and two or more MPEG data streams cannot be reproduced. Therefore, if it is going to reproduce two or more MPEG data streams, the MPEG hardware decoder which became independent to two or more MPEG data streams of each treated simultaneously will be needed, and various inconvenience, such as enlargement of complicatedizing of composition and a device and a steep rise of cost, will be caused.

**[0004]**

[Problem(s) to be Solved by the Invention]As described above, the conventional hardware decoder, If a renewable MPEG data stream tends to reproduce two or more MPEG data streams from being limited to one and being unable to reproduce two or more MPEG streams, The MPEG hardware decoder which became independent to two or more MPEG data streams of each treated simultaneously was needed, and there was a problem of causing various inconvenience, such as enlargement of complicatedizing of composition and a device and a steep rise of cost.

[0005]This invention was made in view of the above-mentioned actual condition, and prepares two or more sets of buffers for making refreshable two or more MPEG data streams for the inside of a hardware decoder, It aims at providing the video decoding device which enabled it to treat two or more video data streams with a single hardware resource.

**[0006]**

[Means for Solving the Problem]This invention inside a system decoder provided in an MPEG hardware decoder, Provide a packet buffer which stores two or more video packets, and inside a video decoder, Two or more registers which store decode information for decoding two or more video streams are provided, and it enabled it to reproduce reproduction of two or more MPEG system streams containing a sound with one hardware resource.

[0007]Namely, two or more packet buffers which store two or more video packets in a system decoder in which this invention treats a video data stream individually, respectively, A means which carries out the output control of the data stored in two or more of these packet buffers synchronizing with a signal supplied from the exterior, A means to notify a state of two or more above-mentioned packet buffers outside is provided, and it enabled it to treat two or more video data streams by a single system decoder.

[0008]This invention builds in two or more registers which store individually two or more sets of decode information corresponding to each data stream for decoding two or more video data

streams in a video decoder treating a video data stream, respectively, It enabled it to perform reproduction of two or more video data streams by a single video decoder.

[0009]Two or more registers which store individually two or more sets of decode information corresponding to each data stream for this invention to decode two or more video data streams to a video decoder treating a video data stream, respectively, Two or more FIFO buffers which store data which is provided corresponding to this register and serves as a candidate for decoding, respectively, Choose one of the above-mentioned registers, and a decoder main part which decodes data stored in a FIFO buffer [ / based on decode information stored in the register concerned ] is provided, It enabled it to perform reproduction of two or more video data streams by a single video decoder.

[0010]A system decoder in which this invention has two or more memory measures which store two or more video data streams individually, respectively in a video decoding device which reproduces a video data stream, A video decoder with two or more memory measures which store individually two or more sets of decode information for reproducing two or more video data streams stored in a memory measure of this system decoder, respectively is provided, Two or more video data streams are renewable in real time with a single hardware resource.

[0011]A system decoder in which this invention has two or more memory measures which store individually two or more video data streams containing sound data, respectively, A video decoder with two or more memory measures which store individually two or more sets of decode information for reproducing two or more video data streams stored in a memory measure of this system decoder, respectively, It is characterized by an MPEG decoder possessing an operating buffer with which decoding including frame data processing of this video decoder is presented, a sound buffer which carries out the reproducing output of the sound data in the above-mentioned data stream synchronizing with video, and a sound decoder.

[0012]

[Embodiment of the Invention]With reference to drawings, one embodiment of this invention is described below. Drawing 1 is a block diagram showing the composition of the MPEG hardware decoder by one embodiment of this invention, and drawing 2 is a block diagram showing the composition of the MPEG decoding system using the MPEG hardware decoder shown in drawing 1.

[0013]The MPEG hardware decoder by one embodiment of this invention shown in drawing 1, The system decoder 10 treating an MPEG data stream [ two or more (here, referred to as four (4)) ], The video decoder 13 which decodes individually four MPEG data streams (4) received from this system decoder 10, respectively, The buffer 15 for decoding which this video decoder 13 uses for the memory of the frame data of I picture, P picture, etc., etc. in a decoding process, The audio buffer 16 which stores temporarily the audio information which accompanies the video in an MPEG data stream, The audio decoder 17 for carrying out the reproducing output of the data stored in this audio buffer 16 and the STC (system time clock) module 18 for taking a reproductive synchronization are constituted as main elements.

[0014]The system decoder 10 which makes the component of the above-mentioned MPEG hardware decoder, the video decoder 13, and the audio buffer 16 are connected via the dedicated bus 102 inside a system, respectively.

[0015]The clock (system time clock) outputted from STC18 via the track 106 is supplied to the system decoder 10, the video decoder 13, and the audio decoder 17.

[0016]Each MPEG data of a maximum of four MPEG data streams (4) read from CD drive 24 shown in drawing 2 is supplied to the system decoder 10 by a packet unit via the system bath 100.

[0017]The system decoder 10 outputs an interrupt signal to the interruption controller 21 shown in drawing 2 via the signal wire 101. A stream selection signal (0-3) is outputted to the video decoder 13 via the signal wire 108.

[0018]Similarly the video decoder 13 sends out a frame identification signal (0-3) to the display controller (VGA) 28 via the signal wire 109 while sending out frame data to the display controller (VGA) 28 shown in drawing 2 via the internal dedicated bus 105.

[0019]Via the internal dedicated bus 103, the video decoder 13 carries out the light of the frame

data of I picture in a decoding process, P picture, etc. to the buffer 15 for decoding, and leads it. The buffer 15 for decoding can store a maximum of eight frame data here.

[0020]The audio decoder 17 sends out the analog audio signal in sync with video to the audio amplifier (AMP) 26 shown in drawing 2 via the signal wire 107. Inside [ above-mentioned ] the system decoder 10, Via the above-mentioned system bath 100. A maximum of four sorts read from CD drive 24 shown in drawing 2. While packet buffer 11-0,11-1 by four FIFO structures of storing individually each MPEG data of (four MPEG data streams) for every stream per video packet, respectively, and 11-2,11-3 are provided, The packet buffer status register 10A shown in four STC counter 12-0,12-1, 12-2,12-3, and drawing 3 along with this packet buffer 11-0,11-1 and 11-2,11-3 is formed.

[0021]. As [ show / in the inside of the above-mentioned system decoder 10 / at drawing 4 (a) - (e) ] The control register 10B for generating various kinds of control information, such as status register information, decoding register information, registration ID acquisition register information, ID registration register information, and ID deletion register information, is formed.

[0022]Inside [ above-mentioned ] the video decoder 13, While four record information register 14-0,14-1 which stores individually four decode information (4) corresponding to each data stream for decoding four MPEG data streams (4), respectively, respectively, and 14-2,14-3 are provided, The data buffer (FIFO) of FIFO structure which stores the data which serves as a candidate for decoding for every data stream, respectively is provided.

[0023]Drawing 2 is a block diagram containing the MPEG hardware decoder shown in above-mentioned drawing 1 showing the composition of an MPEG decoding system, and the numerals 25 in a figure are the MPEG hardware decoders shown in drawing 1 here.

[0024]In drawing 2, 20 is CPU which manages control of the MPEG decoding whole system, and various control information as shown in drawing 4 between the system decoders 10 here is accessed, Each control of a driver's as shown in driver control management's as shown in drawing 5 and drawing 6's, and drawing 8 stream multiplexer processing is managed.

[0025]21 is an interruption controller which generates interruption to CPU20 according to the interrupt request signal received via the interrupt request signal wire 101, The interrupt request signal received from the system decoder 10 of the MPEG hardware decoder 25 shown in above-mentioned drawing 1 here, And according to SCSI interface 23 mentioned later and the interrupt request signal received from the display controller (VGA) 28 grade, interruption is generated to CPU20.

[0026]22 is RAM in which the storing region of the various programs executed by CPU20, workspace, etc. are formed, Here, a stream registration array data field (DA) as shown in drawing 7 which the stream multiplexer driver (DRV) and the stream multiplexer driver (DRV) concerned for performing processing as shown in drawing 6 and drawing 8 use is provided.

[0027]23 is a SCSI interface and sends out here the MPEG data which led from CD drive 24 under control of CPU20 to the MPEG hardware decoder 25 via the system bath 100.

[0028]Two or more MPEG data streams with video stream ID from which 24 is the CD drive connected to SCSI interface 23 via the SCSI cable 110, and each differs here shall be stored.

[0029]As 25 is an MPEG hardware decoder of composition of being shown in above-mentioned drawing 1 and mentioned above, Four packet buffer 11-0,11-1 which stores an MPEG data stream individually, respectively, and the system decoder 10 which built in 11-2,11-3, It has the video decoder 13 which similarly built in four record information register 14-0,14-1 and 14-2,14-3, and two or more MPEG data streams are considered as composition renewable in real time with the single hardware resource.

[0030]26 is an audio amplifier (AMP) which inputs the analog audio signal outputted from the audio decoder 17 provided in the MPEG hardware decoder 25 via the signal wire 107, and carries out audio amplification. 27 is a loudspeaker which outputs the sound (sound) of an audio frequency belt according to the analog audio signal outputted from the audio amplifier (AMP) 26.

[0031]28 is a display controller (VGA) which performs display control under control of CPU20, and carries out display output control of the display screen according to the frame data generated from the MPEG hardware decoder 25 or a software decoder here.

[0032]29 is a monitor which uses various kinds of display screen information as a visible image,

and carries out a display output under the display control of the display controller (VGA) 28, and carries out the display output of the display screen according to the frame data generated from the MPEG hardware decoder 25 or a software decoder here.

[0033]Each system component of above-mentioned CPU20 and the interruption controller 21, RAM22, SCSI interface 23, the MPEG hardware decoder 25, and display controller (VGA) 28 grade via the system bath 100 with an address and a data line. It is connected mutually.

[0034]Via the interrupt request signal wire 101, from SCSI interface 23, the MPEG hardware decoder 25, and display controller (VGA) 28 grade, the interruption controller 21 receives an interrupt signal and sends out an interrupt signal to CPU20 via the interrupt request signal wire 113.

[0035]The MPEG hardware decoder 25 sends out frame data and a frame identification signal (0-3) to the display controller (VGA) 28 via the dedicated bus 105 and the signal wire 115 inside a system.

[0036]SCSI interface 23 inputs the MPEG data by which read access was carried out via the SCSI cable 110 from CD drive 24, and sends it out on the system bath 100 under control of CPU20. The MPEG data sent out on this system bath 100 is sent out to the MPEG hardware decoder 25, and is decoded, or is decoded by the software decoder, and is sent out to the display controller (VGA) 28.

[0037]Drawing 3 is a figure showing the internal register array constitution of the packet buffer status register 10A provided in the system decoder 10 of the above-mentioned MPEG hardware decoder 25. Various kinds of control information shown in drawing 4 on the control register 10B with reference to the contents of this packet buffer status register 10A is generated.

[0038]The register arrangement of this packet buffer status register 10A has four elements, and there is a field which writes in stream ID, operational mode, PTS offset, and a PTS initial value, respectively.

[0039]Drawing 4 is a figure showing the data structure of each control information generated on the control register 10B in the above-mentioned system decoder 10. As a kind of register information which calls register information each control information generated on the control register 10B here, respectively, and can access CPU20. There are status register information, decoded control register information, acquisition register information of registration ID, ID registration register information, ID deletion register information, etc.

[0040]the stream multiplexer driver (DRV) with which drawing 5 was placed on the above-mentioned RAM22 -- control -- dried cuttlefish -- it is a flow chart which shows the control procedure of stream multiplexer processing with the decoder application of a sake.

[0041]Drawing 6 is a flow chart which shows the procedure of the stream multiplexer processing performed by the stream multiplexer driver (DRV) placed on the above-mentioned RAM22.

[0042]Drawing 7 is the figure which was placed on the above-mentioned RAM22 and in which showing the composition of the stream registration array data field (DA) of a stream multiplexer driver (DRV), and receives each four streams here. An array number (0-3), a file name, stream ID, and operational mode are registered, respectively.

[0043]Drawing 8 is a flow chart which shows the flow of the interrupt handler in the above-mentioned stream multiplexer processing. It performs, whenever the interrupt request for notifying the empty (empty) state of packet buffer (FIFO buffer) 11-i (i = 0-3) generated from the system decoder 10 is received by CPU20.

[0044]The figure in which drawing 9 shows the structure (a) of a system stream and the structure (b) of packed data, the figure in which drawing 10 shows the structure of a system header, and drawing 11 are the figures showing the structure of packet data.

[0045]Here explains the operation in the embodiment of this invention with reference to each above-mentioned figure. In this embodiment, it is assumed that video stream ID contained in each differs as conditions for an MPEG system stream renewable by the MPEG hardware decoder 25. Since reproduction speed set renewal of each PTS offset to 10msec about four streams (4) which can be treated by the MPEG hardware decoder 25, it is set to 25fps in this embodiment.

[0046]CPU20 accesses the various control register information shown in drawing 4 from the

system decoder 10. The system decoder 10 in the kind of control register information accessed from CPU20. The status register information which makes structure as shown in drawing 4 (a), the decoded control register information which makes structure as shown in the figure (b), There are acquisition register information of registration ID which makes structure as shown in the figure (c), ID registration register information which makes structure as shown in the figure (d), ID deletion register information which makes structure as shown in the figure (e), etc. On the control register 10B, each of these control register information is generated and is accessed because CPU20 specifies the internal address of the system decoder 10.

[0047]Access of status register information among each above-mentioned control register information, When it is carried out by accessing "0x00", "0x01", "0x02", and "0x03" with the internal address of the system decoder 10 and "0x00" is accessed, When the status to the video stream set as the packet buffer 0 (11-0) is returned and "0x01" is accessed, When the status to the video stream set as the packet buffer 1 (11-1) is returned and "0x02" is accessed, When the status to the video stream set as the packet buffer 2 (11-2) is returned and "0x03" is accessed, the status to the video stream set as the packet buffer 3 (11-3) is returned.

[0048]As shown in drawing 4 (a), there are operation status [A] FIFO empty(nil state of packet buffer 11-i) [B] FIFO full [(full state of packet buffer 11-i) C], stream ID [D], etc. in the item of status register information.

[0049]Operation status [A] is status of stream ID, and "0x03" is set up at the time of "0x02" and Step (top delivery) at the time of "0x00" and Pause (pause) at the time of "0x01" and Stop (stop) at the time of Play (reproduction).

[0050]FIFO empty [B] and FIFO full In the item of [C]. When applicable packet buffer (0-3) 11-0,11-1 and 11-2,11-3 are nil states (empty), "1" is set as FIFO empty [B], and when it is full state (full), "1" is set as FIFOfull [C].

[0051]Access of decoded control register information is performed by accessing "0x04" with the internal address of the system decoder 10. As shown in drawing 4 (b), there are decoded control information [A] and stream ID [B] in the item of decoded control register information, and the same decoded control information as the operation status of status register information is specified as it to the stream specified by stream ID [B].

[0052]Access of the acquisition register information of registration ID is performed by accessing "0x05" with the internal address of the system decoder 10. as shown in drawing 4 (c), there is four stream ID registered in the item of registration ID acquisition register information (a stream -- ID -- zero -- [-- A --] -- a stream -- ID -- one -- [-- B --] -- a stream -- ID -- two -- [-- C --] -- a stream -- ID -- 3 [D]). "0x00" is set to stream ID which is not registered.

[0053]Access of ID registration register information is performed by accessing "0x06" with the internal address of the system decoder 10. As shown in drawing 4 (d), there are an array number [B] and a registration ID value [A] in the item of ID registration register information, and registration ID is assigned to packet buffer 11-i shown with an array number [B].

[0054]Access of ID deletion register information is performed by accessing "0x07" with the internal address of the system decoder 10. As shown in drawing 4 (e), there is an array number [A] in the item of ID deletion register information, and stream ID specified with this array number [A] is deleted.

[0055]From CPU20, if ID registration of the system decoder 10 is carried out by ID registration register information, it will write in registration ID according to the array number specified as the packet buffer status register 10A shown in drawing 3 with the ID registration register.

[0056]At this time, the code with which Stop (stop) is expressed as an initial value is written in the item of operational mode. And packet data with stream ID specified as the same packet buffer 11-i as the array number are stored.

[0057]In this case, the system decoder 10 saves the PTS value which appears first in the packet information in the system stream to decode in the PTS initial value item in the packet buffer status register 10A of drawing 3.

[0058]The aggregate value of this PTS value and PTS offset that were saved, Next, when performing ID registration, if not equivalent as compared with the PTS value of the system stream to register, it will register as it is, and if equivalent, "10" (counted value for 10msec) will

be added to the PTS offset shown with the array number specified with the ID registration register. By comparing this applied value with other saved PTS values, if each is not equivalent, it saves in the PTS offset storing position shown in the array number of drawing 3 by considering that applied value as offset. If equivalent, the same operation as the above will be repeated.

[0059]From CPU20, the system decoder 10 will write in the decoded control information specified as the item of the operational mode of the packet buffer status register 10A specified by stream ID, if decoded control register information is accessed.

[0060]the decoded control information which can be specified here -- Play: "0x01", Stop: "0x00", and Pause : There are four kinds of "0x02" and Step: "0x03." For example, if set as Play: "0x01" from Stop: "0x00", the system decoder 10 will start decoding of the stream as which operational mode was specified.

[0061]When system stream data is inputted into the system decoder 10 via the system bath 100 from CPU20, the system decoder 10, Stream ID of the packet data which divided the data concerned into the video packet and the audio packet, and were divided into the video packet is compared with the stream ID item of the packet buffer status register 10A shown in drawing 3. The video packet data is stored in packet buffer 11-i specified with an array number in agreement.

[0062]The STC counter 12 of the system decoder 10 counts the system time clock inputted from the STC module 18. For example, the value which subtracted the value of the PTS offset 0 from the value of the STC counter 0 (12-0), When PTS/DTS contained in the packet buffer 0 (11-0) becomes equivalent, the system decoder 10, While sending out a stream selection signal (0) to the video decoder 13 via the signal wire 108, the packet data stored in the packet buffer 0 (11-0) are transmitted.

[0063]This operation is performed from the packet buffer 0 to the packet buffer 3 (11-0 to 11-3). When either of each packet buffer 0-3 (11-0,11-1, 11-2,11-3) will be in a nil state (empty), the system decoder 10, An interrupt request signal is sent out to the interruption controller 21 via the signal wire 101, and generating of interruption is notified to the interruption controller 21.

[0064]The interruption controller 21 will report that interruption took place to CPU20 via the signal wire 113, if the notice of an interrupt request signal is received from the MPEG hardware decoder 25.

[0065]CPU20 will start and perform an interrupt handler as shown in drawing 8, if the notice of interruption is received. About this interrupt handler, explanation is added behind.

[0066]The video decoder 13 chooses the decode information used for decoding according to the stream (sent via signal wire 108) selection signal (0-3) specified from the system decoder 10.

[0067]The decode information chosen is one in each decode information stored in four record information register 14-0,14-1 shown in drawing 1, and 14-2,14-3. If the contents of the sequence header contained in a video stream are included in the decode information stored in this record information register 14-0,14-1 and 14-2,14-3 and a sequence header appears in a video stream, those contents will be updated by overwrite.

[0068]The video decoder 13 decodes the video stream information sent from the system decoder 10 using this information. Under the present circumstances, the video decoder 13 uses the buffer 15 for decoding for the storage location of the frame data of I picture in a decoding process, P picture, etc. This buffer 15 for decoding has a size which can store a maximum of eight-frame data.

[0069]The frame data decoded by the video decoder 13 is sent out to the display controller (VGA) 28 via the dedicated bus 105 inside a system. It can come, simultaneously, similarly a frame identification signal (0-3) is sent out to the display controller (VGA) 28 via the signal wire 109.

[0070]Here, the case where one system stream is decoded is considered. Stream ID0 of the video packet registered into the array number 0 with an ID registration register is registered to the system decoder 10.

[0071]Suppose that Play: "0x01" was set as stream ID0 with the decoded control register to the system decoder 10 now. The STC counter 12-0 is reset at this time.

[0072]Since the packet buffer 11-0 is empty (empty), the system decoder 10 sends out an

interrupt request signal to the interruption controller 21 via the signal wire 101, and notifies generating of interruption.

[0073]The interruption controller 21 will report that interruption took place to CPU20, if the above-mentioned interrupt request signal is received. CPU20 will start the interrupt handler in a driver's as shown in drawing 8 stream multiplexer processing, if the notice of the above-mentioned interruption is received.

[0074]As opposed to packet buffer 11-i to which the FIFO empty flag is set with reference to the packet buffer status register 10A in this interrupt handler, The packed data of the file specified by stream ID are transmitted to the system decoder 10 via the system bath 100 (drawing 8 steps 400-403).

[0075]The system decoder 10 divides the inputted pack into a packet, and stores it in packet buffer 11-i specified, respectively with reference to stream ID (streamID) contained in packet data.

[0076]The PTS data which appeared first at this time is stored in the PTS initial value item of the packet buffer status register 10A shown in drawing 3. And the offset compared and searched for by the means mentioned above is stored in the PTS offset 0 of the packet buffer status register 10A shown in drawing 3.

[0077]If the packet data of stream ID0 are stored in packet buffer (0)11-0, STC counter (0)12-0 will begin to count the system time clock from the STC module 18.

[0078]And the value which subtracted the PTS offset 0 from the value of STC counter (0)12-0, Packet buffer (0) When PTS of the video packet data stored in 11-0 and DTS become equivalent, the system decoder 10, The stream selection signal (0) of the stream 0 is notified to the video decoder 13 via the signal wire 108, and from the packet data stored in packet buffer (0)11-0, a video stream is taken out and it transmits to the video decoder 13 via the dedicated bus 102 inside a system.

[0079]If it is an audio packet, the audio stream contained in a packet will be transmitted to the audio buffer 16 via the dedicated bus 102 inside a system. The audio decoder 17 decodes the audio information stream transmitted to the audio buffer 16, and outputs the decoded analog audio signal to the analog signal line 107.

[0080]In the video decoder 13, according to the inputted stream selection signal, record information register 14-0,14-1 and 14-2,14-three to 1 register is chosen, the decode information used for decoding is chosen, and it decodes by the decode information concerned.

[0081]The decoded frame data is outputted on the dedicated bus 105 inside a system, and a frame identification signal is simultaneously outputted to the signal wire 109. The number used when choosing decode information is the same as the number specified by this frame identification signal. For example, the frame data decoded using the decode information 2 of the record information register 14-2 specifies "2" as a frame identification signal.

[0082]Although the case where the above-mentioned operation decoded one system stream was considered, the case where the 2nd system stream is registered here is considered. Stream ID1 of the video packet registered into the array number 1 is registered by ID registration register information to the system decoder 10.

[0083]Although the flow of registration is the same as the case where it mentions above, the calculation work of PTS offset is added. If the stream shown by stream ID1 is stored in the packet buffer 1 (11-1) in the system decoder 10, Comparison with the PTS value which appears first, the already registered PTS initial value, and the aggregate value of PTS offset is performed, and if equivalent, "10" is added to the PTS offset 1. When comparison is all completed, the contents of the PTS offset 1 are determined.

[0084]The timing of transmission of the video stream contained in packet data to the video decoder 13, When the PTS value contained in the packet buffer 1 (11-1) and the value which subtracted the PTS offset 1 from the STC counter 1 (12-1) become equivalent, it is transmitted to the video decoder 13.

[0085]The processing at the time of carrying out supply control of two or more stream data to the MPEG hardware decoder 25 here is explained. The flow of the processing of decoder application to the stream multiplexer driver (DRV) for carrying out supply control of the system

stream to the MPEG hardware decoder 25 is shown in drawing 5.

[0086]Here, if it is setting out of operational mode, the operational mode about the file of a set object (Play/Stop/Pause /Step) will be notified to a stream multiplexer driver (DRV) (drawing 5 step 200,201).

[0087]If it is the open shop operation of a stream file, the stream file name which decodes will be registered to the stream registration array data field (DA) shown in drawing 7 of a stream multiplexer driver (DRV) (drawing 5 step 202,203).

[0088]Under the present circumstances, it becomes registration failure when the stream file names of all four streams are registered into the MPEG hardware decoder 25, Instead of the hardware decoder 25, starting of the software decoder corresponding to the stream by which reproduction instruction was carried out, and reproduction are performed (drawing 5 step 204,205).

[0089]The flow of the above-mentioned stream multiplexer driver's (DRV's) processing is shown in drawing 6. Here, if the notice of setting out of operational mode is received from the decoder application shown in above-mentioned drawing 5, decoded control register information will be accessed to the system decoder 10, and decoded control information will be written in (drawing 6 step 300,306).

[0090]When it is not setting out of operational mode, the file name of the stream notified from decoder application is registered into the stream registration array data field (DA) shown in drawing 7 (drawing 6 step 301).

[0091]Under the present circumstances, when the file name of a stream cannot be registered, an error is notified to decoder application and it ends to it (drawing 6 step 302,303). When the file name of a stream is registered, a stream ID value is registered by taking out stream ID from the file of the notified stream, and accessing ID registration register information to the system decoder 10 (drawing 6 step 304,305).

[0092]The flow of the interrupt handler in the above-mentioned inside of a stream multiplexer is shown in drawing 8. If the interrupt request from the system decoder 10 is received by CPU20, The interrupt handler shown in drawing 8 is performed, the status register information shown in drawing 4 (a) is accessed, the status of the stream registered into the system decoder 10 comes to hand, and the variable I for a count is set as "0" (drawing 8 step 400).

[0093]Next, if the FIFO empty flag of the array number I is checked and the flag concerned stands, packed data will be taken out from the file specified with the array number I, and it will transmit to the system decoder 10 (drawing 8 step 401,402,403). (if it is "1")

[0094]Next, an interrupt handler is ended, when the variable I for a count is ~~\*\*\*\*\*~~<sup>Inclement</sup>,  $I > 3$  is checked and the result serves as truth (drawing 8 step 404,405).

[0095]The structure (figure (a)) of a system stream and the structure (figure (b)) of packed data are shown in drawing 9. A system header (system header()) and packet data (packet()) are contained in a pack.

[0096]The structure of the system header is shown in drawing 10. Stream id in a system header shows video or stream ID of the audio. The structure of packet data is shown in drawing 11. In packet data, PTS (presentation time stamps) and DTS (decoding time stamp), The packet data size (packetdata byte) applicable to a video stream or an audio stream is defined.

[0097]As described above, according to the embodiment of this invention, to the system decoder 10 treating an MPEG data stream. Two or more packet buffer 11-0,11-1 which stores two or more video packets individually, respectively, and 11-2,11-3, The function which carries out the output control of the data stored in this packet buffer 11-0,11-1 and 11-2,11-3 synchronizing with the system time clock supplied from the STC module 18 is provided, Two or more record information register 14-0,14-1 which stores individually two or more sets of decode information corresponding to each data stream for decoding two or more data streams in the video decoder 13, respectively, and 14-2,14-3, It corresponds to this register 14-0,14-1 and 14-2,14-3, Two or more data buffers (FIFO) which store the data used as the candidate for decoding, Choose one of the above-mentioned registers, and the decoder main part which decodes the data stored in the data buffer [ / based on the decode information stored in the register 14-i concerned ] (FIFO) is provided, By having had composition which can treat two or more video data streams

by a single hardware decoder. When realizing the MPEG hardware decoder which can treat two or more MPEG data streams in parallel, It can realize cheaply and compactly with easy composition, without providing the MPEG hardware decoder which became independent to two or more MPEG data streams of each treated simultaneously.

[0098]

[Effect of the Invention]By having provided two or more sets of buffers and two or more stream control facilities for making refreshable two or more MPEG data streams in the inside of a hardware decoder according to this invention, as a full account was given above, The video decoding device which can reproduce two or more video data streams in real time with a single hardware resource can be provided.

[0099]Namely, two or more packet buffers which store two or more video packets in the system decoder treating a video data stream individually, respectively according to this invention, By having had composition possessing the means which carries out the output control of the data stored in two or more of these packet buffers synchronizing with the signal supplied from the exterior, and a means to notify the state of two or more above-mentioned packet buffers outside, Two or more video data streams are renewable in real time by a single system decoder.

[0100]To the video decoder which treats a video data stream according to this invention. By having had composition which builds in two or more registers which store individually two or more sets of decode information corresponding to each data stream for decoding two or more video data streams, respectively, reproduction of two or more video data streams can be performed by a single video decoder.

[0101]To the video decoder which treats a video data stream according to this invention. Two or more registers which store individually two or more sets of decode information corresponding to each data stream for decoding two or more video data streams, respectively, Two or more FIFO buffers which store the data which is provided corresponding to this register and serves as a candidate for decoding, respectively, By having chosen one of the above-mentioned registers, and having had composition possessing the decoder main part which decodes the data stored in the FIFO buffer [ / based on the decode information stored in the register concerned ],

Reproduction of two or more video data streams can be performed by a single video decoder.

[0102]The system decoder which has two or more memory measures which store two or more video data streams individually, respectively in the video decoding device which reproduces a video data stream according to this invention, By having had composition possessing a video decoder with two or more memory measures which store individually two or more sets of decode information for reproducing two or more video data streams stored in the memory measure of this system decoder, respectively, Two or more video data streams are renewable in real time with a single hardware resource.

[0103]A system decoder with two or more memory measures which store individually two or more video data streams containing sound data, respectively according to this invention, A video decoder with two or more memory measures which store individually two or more sets of decode information for reproducing two or more video data streams stored in the memory measure of this system decoder, respectively, The operating buffer with which decoding including frame data processing of this video decoder is presented, By having had the sound buffer and sound decoder which carry out the reproducing output of the sound data in the above-mentioned data stream synchronizing with video, and having constituted the MPEG decoder, two or more video data streams are renewable in real time with a single hardware resource.

---

[Translation done.]

**\* NOTICES \***

JP0 and INPIT are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.\*\*\*\* shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

---

**DESCRIPTION OF DRAWINGS**

---

**[Brief Description of the Drawings]**

**[Drawing 1]**The block diagram showing the composition of the MPEG hardware decoder by the embodiment of this invention.

**[Drawing 2]**The block diagram showing the composition of the MPEG decoding system containing the MPEG hardware decoder shown in drawing 1.

**[Drawing 3]**The figure showing the internal register array constitution of the packet buffer status register 10A provided in the system decoder 10 of the MPEG hardware decoder 25 in the above-mentioned embodiment.

**[Drawing 4]**The figure showing the data structure of each control information generated on the control register 10B in the system decoder 10 in the above-mentioned embodiment.

**[Drawing 5]**the stream multiplexer driver (DRV) in the above-mentioned embodiment placed on RAM22 -- control -- dried cuttlefish -- the flow chart which shows the control procedure of stream multiplexer processing with the decoder application of a sake.

**[Drawing 6]**The flow chart which shows the procedure of the stream multiplexer processing performed by the stream multiplexer driver (DRV) in the above-mentioned embodiment placed on RAM22.

**[Drawing 7]**The figure showing the composition of the stream registration array data field (DA) of the stream multiplexer driver (DRV) placed on RAM22 in the above-mentioned embodiment.

**[Drawing 8]**The flow chart which shows the flow of the interrupt handler in stream multiplexer processing in the above-mentioned embodiment.

**[Drawing 9]**The figure showing the structure (a) of a system stream, and the structure (b) of packed data.

**[Drawing 10]**The figure showing the structure of a system header.

**[Drawing 11]**The figure showing the structure of packet data.

**[Description of Notations]**

10 -- A system decoder, 10A -- Packet buffer status register, 10B -- Control register and 11--0,11-1, 11-2,11-3 -- Packet buffer, 12-0,12-1, 12-2,12-3 -- An STC counter, 13 -- Video decoder, 14-0,14-1, 14-2,14-3 -- Record information register, 15 -- The buffer for decoding, 16 -- An audio buffer, 17 -- Audio decoder, 18 -- An STC (system time clock) module, 20 -- CPU, 21 -- An interruption controller, 22 -- RAM, 23 -- SCSI interface, 24 [ -- A loudspeaker, 28 / -- A display controller (VGA) 29 / -- A monitor, DRV / -- A stream multiplexer driver, DA / -- Stream registration array data field. ] -- A CD drive, 25 -- An MPEG hardware decoder, 26 -- An audio amplifier (AMP), 27

---

[Translation done.]

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-93577

(43)公開日 平成9年(1997)4月4日

(51) Int.Cl.<sup>6</sup> 識別記号 庁内整理番号 F I 技術表示箇所  
 H 0 4 N 7/24 H 0 4 N 7/13 Z  
 5/92 9382-5K H 0 3 M 7/00  
 7/08 H 0 4 N 5/92 H  
 7/081 7/08 Z  
 // H 0 3 M 7/00

審査請求 未請求 請求項の数8 OL (全 13 頁)

(21)出願番号 特願平7-251253

(22)出願日 平成7年(1995)9月28日

(71) 出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72)発明者 芝原 昭彦

東京都青梅市末広町2丁目9番地 株式会社東芝青梅工場内

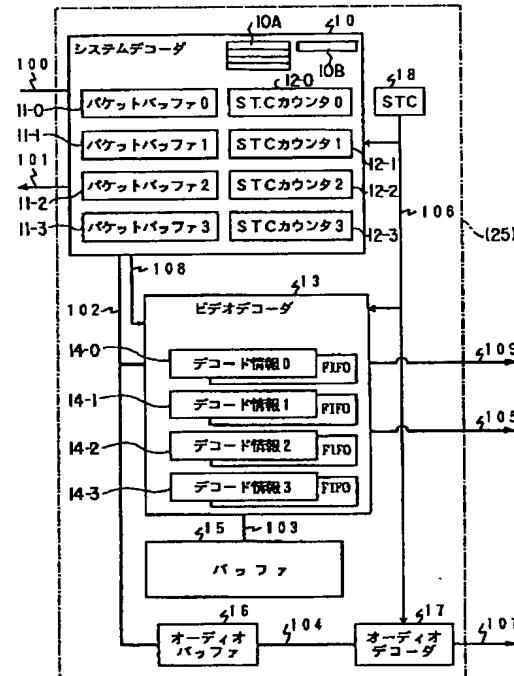
(74)代理人 爾理士 鈴江 武彥

(54) [発明の名称] 動画像デコード装置

(57) 【要約】

【課題】本発明は、単一のMPEGハードウェアデコーダにより複数のMPEGデータストリームを扱うことができる構成としたことを特徴とする。

【解決手段】MPEGデータストリームを扱うシステムデコーダ10に、複数個のパケットバッファ11-0, 11-1, …と、このパケットバッファに貯えられたデータをSTCモジュール18より供給されるシステムタイムクロックに同期して出力制御する機能とを設け、ビデオデコーダ13に、複数のデータストリームに対応する複数組のデコード情報をそれぞれ個別に格納する複数個のレコード情報レジスタ14-0, 14-1, …と、このレジスタに対応して、デコード対象となるデータを貯える複数個のデータバッファ(FIFO)とを設けて、单一のハードウェアデコーダにより複数の動画像データストリームを扱うことができる構成としたことを特徴とする。



1

## 【特許請求の範囲】

【請求項1】 複数のビデオパケットをそれぞれ個別に貯える複数個のパケットバッファと、この複数個のパケットバッファに貯えられたデータを外部より供給される信号に同期して出力制御する手段と、上記複数個のパケットバッファの状態を外部に通知する手段とを具備し、単一のハードウェアリソースで複数の動画像データストリームを扱うことができるようとしたことを特徴とするシステムデコーダ。

【請求項2】 システムストリームの中に含まれているストリームIDを識別して格納先パケットバッファを決定する手段をもつ請求項1記載のシステムデコーダ。

【請求項3】 複数個のパケットバッファそれぞれに対応して、データストリーム中に含まれる同期データを更新制御する手段をもつ請求項1記載のシステムデコーダ。

【請求項4】 複数個のパケットバッファを構成する複数個のFIFOバッファと、このFIFOバッファに対応して設けられた同期制御のためのSTCカウンタと、上記FIFOバッファのデータを管理する制御レジスタとを内蔵してなる請求項1記載のシステムデコーダ。

【請求項5】 複数の動画像データストリームをデコードするための各データストリームに対応する複数組のコード情報をそれぞれ個別に格納する複数のレジスタを内蔵して、複数の動画像データストリームの再生を单一のハードウェアリソースで実行できるようにしたことを特徴とするビデオデコーダ。

【請求項6】 複数の動画像データストリームをデコードするための各データストリームに対応する複数組のコード情報をそれぞれ個別に格納する複数個のレジスタと、

このレジスタに対応して設けられ、それぞれデコード対象となるデータを貯えるFIFOバッファと、上記レジスタの1つを選択し、当該レジスタに貯えられたデコード情報をもとに對応するFIFOバッファに貯えられたデータをデコード処理するデコーダ本体とを具備して、複数の動画像データストリームの再生を单一のハードウェアリソースで実行できるようにしたことを特徴とするビデオデコーダ。

【請求項7】 複数の動画像データストリームをそれぞれ個別に貯える複数の記憶手段をもつシステムデコーダと、このシステムデコーダの記憶手段に貯えられた複数の動画像データストリームを再生するための複数組のデコード情報をそれぞれ個別に貯える複数の記憶手段をもつビデオデコーダとを具備して、単一のハードウェアリソースで複数の動画像データストリームをリアルタイムに再生できることを特徴とする動画像デコード装置。

【請求項8】 サウンドデータを含む複数の動画像データストリームをそれぞれ個別に貯える複数の記憶手段をもつシステムデコーダと、このシステムデコーダの記憶

2

手段に貯えられた複数の動画像データストリームを再生するための複数組のデコード情報をそれぞれ個別に貯える複数の記憶手段をもつビデオデコーダと、このビデオデコーダのフレームデータ処理を含むデコード処理に供される作業用バッファと、上記データストリーム中のサウンドデータを動画像に同期して再生出力するサウンドバッファ及びサウンドデコーダとを具備してなることを特徴とするMPEGデコーダ。

## 【発明の詳細な説明】

10 【0001】

【発明の属する技術分野】本発明は、複数種の動画像データストリーム（例えばMPEGシステムストリーム）の再生機能を单一ハードウェアリソースにより実現可能にした動画像デコード装置に関する。

【0002】

【従来の技術】MPEGで圧縮された動画像データを再生するためのMPEGハードウェアデコーダの代表的なものとして、SIGMA DESIGNS社製のReal Magicがある。このハードウェアデコーダの性能は、 $352 \times 240$ ドットサイズのカラー動画像を 30fps (frame per second) の速度でデコードすることができる。

20

【0003】しかしながら、上記した従来のハードウェアデコーダに於いては、再生できるMPEGデータストリームが1つ（1本）に限られ、複数のMPEGデータストリームを再生することはできない。従って複数のMPEGデータストリームを再生しようとすると、同時に扱う複数のMPEGデータストリームそれぞれに対して独立したMPEGハードウェアデコーダが必要となり、構成の繁雑化、装置の大型化、及びコストの大幅な上昇等、種々の不都合を招く。

【0004】

【発明が解決しようとする課題】上記したように、従来のハードウェアデコーダは、再生できるMPEGデータストリームが1つに限定され複数のMPEGストリームを再生することができないことから、複数のMPEGデータストリームを再生しようとすると、同時に扱う複数のMPEGデータストリームそれぞれに対して独立したMPEGハードウェアデコーダが必要となり、構成の繁雑化、装置の大型化、及びコストの大幅な上昇等、種々の不都合を招くという問題があった。

30

【0005】本発明は上記実情に鑑みなされたもので、ハードウェアデコーダ内部に複数のMPEGデータストリームを再生可能にするための複数組のバッファを用意して、単一のハードウェアリソースで複数の動画像データストリームを扱うことができるようとした動画像デコード装置を提供することを目的とする。

【0006】

【課題を解決するための手段】本発明は、MPEGハードウェアデコーダに設けられたシステムデコーダの内部に、複数のビデオパケットを格納するパケットバッファ

50

を設け、ビデオデコーダの内部に、複数のビデオストリームをデコードするためのデコード情報を格納する複数のレジスタを設けて、音声を含む複数のMPEGシステムストリームの再生を1つのハードウェアリソースで再生することができるようとしたことを特徴とする。

【0007】即ち本発明は、動画像データストリームを扱うシステムデコーダに、複数のビデオパケットをそれぞれ個別に貯える複数個のパケットバッファと、この複数個のパケットバッファに貯えられたデータを外部より供給される信号に同期して出力制御する手段と、上記複数個のパケットバッファの状態を外部に通知する手段とを具備して、単一のシステムデコーダで複数の動画像データストリームを扱うことができるようにしたことを特徴とする。

【0008】又、本発明は、動画像データストリームを扱うビデオデコーダに、複数の動画像データストリームをデコードするための各データストリームに対応する複数組のデコード情報をそれぞれ個別に格納する複数のレジスタを内蔵して、複数の動画像データストリームの再生を単一のビデオデコーダで実行できることを特徴とする。

【0009】又、本発明は、動画像データストリームを扱うビデオデコーダに、複数の動画像データストリームをデコードするための各データストリームに対応する複数組のデコード情報をそれぞれ個別に格納する複数個のレジスタと、このレジスタに対応して設けられ、それぞれデコード対象となるデータを貯える複数個のFIFOバッファと、上記レジスタの1つを選択し、当該レジスタに貯えられたデコード情報をもとに對応するFIFOバッファに貯えられたデータをデコード処理するデコーダ本体とを具備して、複数の動画像データストリームの再生を単一のビデオデコーダで実行できることを特徴とする。

【0010】又、本発明は、動画像データストリームを再生する動画像デコード装置に於いて、複数の動画像データストリームをそれぞれ個別に貯える複数の記憶手段をもつシステムデコーダと、このシステムデコーダの記憶手段に貯えられた複数の動画像データストリームを再生するための複数組のデコード情報をそれぞれ個別に貯える複数の記憶手段をもつビデオデコーダとを具備して、単一のハードウェアリソースで複数の動画像データストリームをリアルタイムに再生できることを特徴とする。

【0011】又、本発明は、サウンドデータを含む複数の動画像データストリームをそれぞれ個別に貯える複数の記憶手段をもつシステムデコーダと、このシステムデコーダの記憶手段に貯えられた複数の動画像データストリームを再生するための複数組のデコード情報をそれぞれ個別に貯える複数の記憶手段をもつビデオデコーダと、このビデオデコーダのフレームデータ処理を含むデ

コード処理に供される作業用バッファと、上記データストリーム中のサウンドデータを動画像に同期して再生出力するサウンドバッファ及びサウンドデコーダとを具備してなるMPEGデコーダを特徴とする。

#### 【0012】

【発明の実施の形態】以下図面を参照して本発明の一実施形態を説明する。図1は本発明の一実施形態によるMPEGハードウェアデコーダの構成を示すブロック図であり、図2は図1に示すMPEGハードウェアデコーダを用いたMPEGデコードシステムの構成を示すブロック図である。

【0013】図1に示す、この発明の一実施形態による、MPEGハードウェアデコーダは、複数（ここでは4つ（4本）とする）のMPEGデータストリームを扱うシステムデコーダ10と、このシステムデコーダ10より受ける4つ（4本）のMPEGデータストリームをそれぞれ個別にデコード処理するビデオデコーダ13と、このビデオデコーダ13がデコード過程に於いてIピクチャ、Pピクチャ等のフレームデータの一時記憶等に使用するデコード用バッファ15と、MPEGデータストリーム中の動画像に付随するオーディオデータを一時記憶するオーディオバッファ16と、このオーディオバッファ16に貯えられたデータを再生出力するためのオーディオデコーダ17と、再生の同期をとるためのSTC（システムタイムクロック）モジュール18とを主な要素として構成される。

【0014】上記MPEGハードウェアデコーダの構成要素をなすシステムデコーダ10と、ビデオデコーダ13と、オーディオバッファ16とはそれぞれシステム内部の専用バス102を介して接続される。

【0015】システムデコーダ10、ビデオデコーダ13、及びオーディオデコーダ17には、線路106を介してSTC18より出力されるクロック（システムタイムクロック）が供給される。

【0016】システムデコーダ10には、システムバス100を介して、図2に示すCDドライブ24から読出された最大4つ（4本）のMPEGデータストリームの各MPEGデータがパケット単位で供給される。

【0017】システムデコーダ10は、信号線101を介して、図2に示す割り込みコントローラ21に割り込み信号を出力する。又、信号線108を介して、ビデオデコーダ13にストリーム選択信号（0～3）を出力する。

【0018】ビデオデコーダ13は、内部の専用バス105を介して図2に示す表示コントローラ（VGA）28にフレームデータを送出するとともに、信号線109を介して同じく表示コントローラ（VGA）28にフレーム識別信号（0～3）を送出する。

【0019】又、ビデオデコーダ13は、デコード処理過程に於けるIピクチャ、Pピクチャ等のフレームデータ

5

タを内部の専用バス103を介してデコード用バッファ15にライトしリードする。尚、デコード用バッファ15は、ここでは最大8つのフレームデータを格納することができる。

【0020】オーディオデコーダ17は信号線107を介して図2に示すオーディオ増幅器(AMP)26に動画像に同期したアナログオーディオ信号を送出する。上記したシステムデコーダ10の内部には、上記システムバス100を介して図2に示すCDドライブ24から読み出された最大4種(4本)のMPEGデータストリームの各MPEGデータをビデオパケット単位でそれぞれストリーム毎に個別に貯える4個の FIFO構造によるパケットバッファ11-0, 11-1, 11-2, 11-3が設けられるとともに、この各パケットバッファ11-0, 11-1, 11-2, 11-3に付随して4個のSTCカウンタ12-0, 12-1, 12-2, 12-3、及び図3に示すパケットバッファステータスレジスタ10Aが設けられる。

【0021】更に上記システムデコーダ10の内部には、図4(a)~(e)に示すような、ステータスレジスタ情報、デコードレジスタ情報、登録ID入手レジスタ情報、ID登録レジスタ情報、ID削除レジスタ情報等の各種の制御情報を生成するための制御レジスタ10Bが設けられる。

【0022】又、上記したビデオデコーダ13の内部には、4つ(4本)のMPEGデータストリームをそれぞれデコードするための各データストリームに対応する4つ(4本)のデコード情報をそれぞれ個別に格納する4個のレコード情報レジスタ14-0, 14-1, 14-2, 14-3が設けられるとともに、各データストリーム毎にそれぞれデコード対象となるデータを貯えるFIFO構造のデータバッファ(FIFO)が設けられる。

【0023】図2は上記した図1に示すMPEGハードウェアデコーダを含む、MPEGデコードシステムの構成を示すブロック図であり、ここでは図中の符号25が図1に示したMPEGハードウェアデコーダである。

【0024】図2に於いて、20はMPEGデコードシステム全体の制御を司るCPUであり、ここではシステムデコーダ10との間で図4に示すような各種制御情報をアクセスして、図5に示すようなドライバ制御処理、及び、図6、図8に示すようなドライバのストリームマルチプレクサ処理の各制御を司る。

【0025】21は割り込み要求信号線101を介して受けた割り込み要求信号に従いCPU20に対して割り込みを発生する割り込みコントローラであり、ここでは上記1に示すMPEGハードウェアデコーダ25のシステムデコーダ10から受けた割り込み要求信号、及び後述するSCSIインターフェース23、表示コントローラ(VGA)28等から受けた割り込み要求信号に従い、CPU20に対して割り込みを発生する。

【0026】22はCPU20により実行される各種プ

6

ログラムの格納領域、作業領域等が形成されるRAMであり、ここでは図6及び図8に示すような処理を行なうためのストリームマルチプレクサドライバ(DRV)、及び当該ストリームマルチプレクサドライバ(DRV)が使用する図7に示すようなストリーム登録配列データ領域(DA)が設けられる。

【0027】23はSCSIインターフェースであり、ここではCPU20の制御の下に、CDドライブ24よりリードしたMPEGデータをシステムバス100を介してMPEGハードウェアデコーダ25に送出する。

【0028】24はSCSIケーブル110を介してSCSIインターフェース23に接続されたCDドライブであり、ここではそれが異なるビデオストリームIDをもつ複数のMPEGデータストリームが格納されているものとする。

【0029】25は上記した図1に示す構成のMPEGハードウェアデコーダであり、上述したように、MPEGデータストリームをそれぞれ個別に貯える4つのパケットバッファ11-0, 11-1, 11-2, 11-3を内蔵したシステムデコーダ10と、同じく4つのレコード情報レジスタ14-0, 14-1, 14-2, 14-3を内蔵したビデオデコーダ13とを有して、単一のハードウェアリンクで複数のMPEGデータストリームをリアルタイムに再生できる構成としている。

【0030】26はMPEGハードウェアデコーダ25に設けられたオーディオデコーダ17より出力されるアナログオーディオ信号を信号線107を介して入力しオーディオ増幅するオーディオ増幅器(AMP)である。

27はオーディオ増幅器(AMP)26より出力されたアナログオーディオ信号に従う可聴周波数帯の音声(サウンド)を出力するスピーカである。

【0031】28はCPU20の制御の下に表示制御を行なう表示コントローラ(VGA)であり、ここではMPEGハードウェアデコーダ25、又はソフトウェアデコーダより生成されるフレームデータに従う表示画面を表示出力制御する。

【0032】29は表示コントローラ(VGA)28の表示制御の下に各種の表示画面情報を可視画像として表示出力するモニタであり、ここではMPEGハードウェアデコーダ25、又はソフトウェアデコーダより生成されるフレームデータに従う表示画面を表示出力する。

【0033】上記したCPU20、割り込みコントローラ21、RAM22、SCSIインターフェース23、MPEGハードウェアデコーダ25、及び表示コントローラ(VGA)28等の各システム構成要素はアドレス及びデータラインをもつシステムバス100を介して相互に接続される。

【0034】割り込みコントローラ21は、割り込み要求信号線101を介してSCSIインターフェース23、MPEGハードウェアデコーダ25、表示コントローラ

(VGA) 28等より割り込み信号を受付け、割り込み要求信号線113を介してCPU20に割り込み信号を送出する。

【0035】又、MPEGハードウェアデコーダ25は、システム内部の専用バス105及び信号線115を介して表示コントローラ(VGA)28にフレームデータ及びフレーム識別信号(0~3)を送出する。

【0036】SCSIインターフェース23は、CDドライブ24よりリードアクセスされたMPEGデータをSCSIケーブル110を介して入力し、CPU20の制御の下に、システムバス100上に送出する。このシステムバス100上に送出されたMPEGデータはMPEGハードウェアデコーダ25に送出されてデコードされ、又はソフトウェアデコーダによりデコードされて、表示コントローラ(VGA)28に送出される。

【0037】図3は上記MPEGハードウェアデコーダ25のシステムデコーダ10に設けられたパケットバッファステータスレジスタ10Aの内部レジスタ配列構成を示す図である。このパケットバッファステータスレジスタ10Aの内容を参照して制御レジスタ10B上に図4に示す各種の制御情報が生成される。

【0038】このパケットバッファステータスレジスタ10Aのレジスタ配列は4つの要素を持ち、それぞれ、ストリームID、動作モード、PTSオフセット、PTS初期値を書き込む領域がある。

【0039】図4は上記システムデコーダ10内の制御レジスタ10B上に生成される各制御情報のデータ構造を示す図であり、ここでは制御レジスタ10B上に生成される各制御情報をそれぞれレジスタ情報と称し、CPU20がアクセスできるレジスタ情報の種類として、ステータスレジスタ情報、デコード制御レジスタ情報、登録IDの入手レジスタ情報、ID登録レジスタ情報、ID削除レジスタ情報等がある。

【0040】図5は上記RAM22上に置かれたストリームマルチプレクサドライバ(DRV)を制御するためのデコーダアプリケーションでのストリームマルチプレクサ処理の制御手順を示すフローチャートである。

【0041】図6は上記RAM22上に置かれたストリームマルチプレクサドライバ(DRV)により実行されるストリームマルチプレクサ処理の手順を示すフローチャートである。

【0042】図7は上記RAM22上に置かれた、ストリームマルチプレクサドライバ(DRV)のストリーム登録配列データ領域(DA)の構成を示す図であり、ここでは4つの各ストリームに対して、それぞれ配列番号(0~3)、ファイル名、ストリームID、動作モードが登録される。

【0043】図8は上記ストリームマルチプレクサ処理に於ける割り込み処理ルーチンの流れを示すフローチャートであり、システムデコーダ10から発生される、バ

ケットバッファ(FIFOバッファ)11-i(i=0~3)の空(empty)状態を通知するための割り込み要求がCPU20に受付けられる度に実行される。

【0044】図9はシステムストリームの構造(a)と、パックデータの構造(b)を示す図、図10はシステムヘッダの構造を示す図、図11はパケットデータの構造を示す図である。

【0045】ここで上記各図を参照して本発明の実施形態に於ける動作を説明する。この実施形態に於いては、MPEGハードウェアデコーダ25で再生できるMPEGシステムストリームの条件として、それぞれに含まれるビデオストリームIDが異なっているものとする。また再生速度は、MPEGハードウェアデコーダ25で扱うことのできる4つ(4本)のストリームについてそれぞれのPTSオフセットの更新を10msecとしたことから、この実施形態に於いては25fpsとする。

【0046】CPU20は、システムデコーダ10から、図4に示す各種制御レジスタ情報をアクセスする。システムデコーダ10がCPU20からアクセスされる制御レジスタ情報の種類には、図4(a)に示すような構造をなすステータスレジスタ情報、同図(b)に示すような構造をなすデコード制御レジスタ情報、同図(c)に示すような構造をなす登録IDの入手レジスタ情報、同図(d)に示すような構造をなすID登録レジスタ情報、及び同図(e)に示すような構造をなすID削除レジスタ情報等がある。これらの各制御レジスタ情報はCPU20がシステムデコーダ10の内部アドレスを指定することで制御レジスタ10B上に生成されアクセスされる。

【0047】上記各制御レジスタ情報のうち、ステータスレジスタ情報のアクセスは、システムデコーダ10の内部アドレスで“0x00”, “0x01”, “0x02”, “0x03”をアクセスすることで行なわれ、“0x00”をアクセスしたときは、パケットバッファ0(11-0)に設定されているビデオストリームに対してのステータスを返し、“0x01”をアクセスしたときは、パケットバッファ1(11-1)に設定されているビデオストリームに対してのステータスを返し、“0x02”をアクセスしたときは、パケットバッファ2(11-2)に設定されているビデオストリームに対してのステータスを返し、“0x03”をアクセスしたときは、パケットバッファ3(11-3)に設定されているビデオストリームに対してのステータスを返す。

【0048】ステータスレジスタ情報の項目には、図4(a)に示すように、動作ステータス[A]、FIFO empty(パケットバッファ11-iの空状態)[B]、FIFO full(パケットバッファ11-iの満状態)[C]、及びストリームID[D]等がある。

【0049】動作ステータス[A]は、ストリームIDのステータスであり、Play(再生)のとき“0x0

1”、Stop（停止）のとき“0x00”、Pause（休止）のとき“0x02”、Step（コマ送り）のとき“0x03”が設定されている。

【0050】 FIFO empty [B]、FIFO full [C] の項目には、該当するパケットバッファ（0～3）11-0, 11-1, 11-2, 11-3が空状態（empty）のとき、FIFO empty [B] に“1”が設定され、満状態（full）のとき、FIFO full [C] に“1”が設定される。

【0051】デコード制御レジスタ情報のアクセスは、システムデコーダ10の内部アドレスで“0x04”をアクセスすることにより行なわれる。デコード制御レジスタ情報の項目には、図4（b）に示すように、デコード制御情報[A]とストリームID[B]とがあり、ストリームID[B]で指定されるストリームに対してステータスレジスタ情報の動作ステータスと同じデコード制御情報を指定する。

【0052】登録IDの入手レジスタ情報のアクセスは、システムデコーダ10の内部アドレスで“0x05”をアクセスすることで行なわれる。登録ID入手レジスタ情報の項目には、図4（c）に示すように、登録されているストリームIDが4つある（ストリームID 0[A]、ストリームID 1[B]、ストリームID 2[C]、ストリームID 3[D]）。登録されていないストリームIDには“0x00”が設定されている。

【0053】ID登録レジスタ情報のアクセスは、システムデコーダ10の内部アドレスで“0x06”をアクセスすることにより行なわれる。ID登録レジスタ情報の項目には、図4（d）に示すように、配列番号[B]と登録ID値[A]があり、配列番号[B]で示されるパケットバッファ11-iに登録IDを割り当てる。

【0054】ID削除レジスタ情報のアクセスは、システムデコーダ10の内部アドレスで“0x07”をアクセスすることにより行なわれる。ID削除レジスタ情報の項目には、図4（e）に示すように、配列番号[A]があり、この配列番号[A]で指定されるストリームIDが削除される。

【0055】システムデコーダ10は、CPU20から、ID登録レジスタ情報によりID登録されると、図3に示すパケットバッファステータスレジスタ10Aに、そのID登録レジスタで指定された配列番号に従う登録IDを書き込む。

【0056】このとき、動作モードの項目には、初期値としてStop（停止）を表わすコードが書き込まれる。そして、その配列番号と同じパケットバッファ11-iに、指定されたストリームIDを持つパケットデータを格納する。

【0057】更にこの際、システムデコーダ10は、デコードするシステムストリームにあるパケット情報で最初に現れるPTS値を図3のパケットバッファステータ

スレジスタ10AにあるPTS初期値項目に保存しておく。

【0058】この保存されたPTS値とPTSオフセットとの加算値は、次にID登録を行なうときに、登録するシステムストリームのPTS値と比較して同値でなければそのまま登録をし、同値であればID登録レジスタで指定された配列番号で示すPTSオフセットに「10」（10 msec分のカウント値）を加える。この加えた値と他の保存されたPTS値を比較してそれが同値でなければ、その加えた値をオフセットとして図3の配列番号に示されるPTSオフセット格納位置に保存する。又、同値であれば上記同様の動作を繰り返す。

【0059】システムデコーダ10は、CPU20から、デコード制御レジスタ情報がアクセスされると、ストリームIDで指定されるパケットバッファステータスレジスタ10Aの動作モードの項目に、指定されたデコード制御情報を書き込む。

【0060】ここで指定できるデコード制御情報には、Play：“0x01”、Stop：“0x00”、Pause：“0x02”、Step：“0x03”の4種類がある。例えば、Stop：“0x00”から、Play：“0x01”に設定されると、システムデコーダ10は、動作モードが指定されたストリームのデコード処理を開始する。

【0061】CPU20からシステムバス100を介してシステムデコーダ10にシステムストリームデータが入力されると、システムデコーダ10は、当該データをビデオパケットとオーディオパケットとに分け、ビデオパケットに分けられたパケットデータのストリームIDを図3に示すパケットバッファステータスレジスタ10AのストリームID項目と比較して、一致するところの配列番号で指定されるパケットバッファ11-iに、そのビデオパケットデータを格納する。

【0062】システムデコーダ10のSTCカウンタ12は、STCモジュール18から入力されるシステムタイムクロックをカウントする。例えば、STCカウンタ0（12-0）の値からPTSオフセット0の値を減算した値と、パケットバッファ0（11-0）に含まれるPTS/DTSが同値になると、システムデコーダ10は、ビデオデコーダ13に、信号線108を介してストリーム選択信号（0）を送出するとともに、パケットバッファ0（11-0）に格納されているパケットデータを転送する。

【0063】この動作をパケットバッファ0からパケットバッファ3（11-0～11-3）まで行なう。尚、それぞれのパケットバッファ0～3（11-0, 11-1, 11-2, 11-3）のいずれかが空状態（empty）になると、システムデコーダ10は、信号線101を介して割り込みコントローラ21に割り込み要求信号を出し、割り込みの発生を割り込みコントローラ21に通知する。

【0064】割り込みコントローラ21は、MPEGハ

11

ードウェアデコーダ25から割り込み要求信号の通知を受けると、信号線113を介しCPU20に対して、割り込みが起こったことを通知する。

【0065】CPU20は、割り込みの通知を受けると、図8に示すような割り込み処理ルーチンを起動し実行する。この割り込み処理ルーチンに関しては後に説明を加える。

【0066】ビデオデコーダ13は、システムデコーダ10から指定された（信号線108を介して送られた）ストリーム選択信号（0～3）に従って、デコード処理に使用するデコード情報を選択する。

【0067】選択されるデコード情報は、図1に示す4つのレコード情報レジスタ14-0, 14-1, 14-2, 14-3に貯えられた各デコード情報の中の1つである。このレコード情報レジスタ14-0, 14-1, 14-2, 14-3に貯えられたデコード情報には、ビデオストリームに含まれるシーケンスヘッダの内容が含まれ、シーケンスヘッダが、ビデオストリーム中に現れると、その内容が上書きで更新される。

【0068】ビデオデコーダ13は、この情報をを利用してシステムデコーダ10より送られてくるビデオストリーム情報をデコードする。この際、ビデオデコーダ13は、デコード過程でのIピクチャ、Pピクチャ等のフレームデータの格納先にデコード用バッファ15を使用する。このデコード用バッファ15は、最大8フレームデータを格納できる大きさをもつ。

【0069】ビデオデコーダ13でデコードされたフレームデータは、システム内部の専用バス105を介して表示コントローラ（VGA）28に送出される。これと同時に、信号線109を介してフレーム識別信号（0～3）が同じく表示コントローラ（VGA）28に送出される。

【0070】ここで、1つのシステムストリームをデコードする場合を考える。システムデコーダ10に対して、ID登録レジスタによって、配列番号0に登録するビデオパケットのストリームID0を登録する。

【0071】いまシステムデコーダ10に対してデコード制御レジスタにより、ストリームID0に、Play：“0x01”が設定されたとする。このとき、STCカウンタ12-0がリセットされる。

【0072】システムデコーダ10は、パケットバッファ11-0が空（empty）なので、信号線101を介して割り込み要求信号を割り込みコントローラ21に送出し割り込みの発生を通知する。

【0073】割り込みコントローラ21は、上記割り込み要求信号を受けると、CPU20に対して割り込みが起こったことを通知する。CPU20は、上記割り込みの通知を受けると、図8に示すような、ドライバのストリームマルチプレクサ処理に於ける割り込み処理ルーチンを起動する。

12

【0074】この割り込み処理ルーチンでは、パケットバッファステータスレジスタ10Aを参照し、 FIFO emptyフラグがセットされているパケットバッファ11-iに対して、ストリームIDで指定されるファイルのパックデータをシステムバス100を介し、システムデコーダ10に転送する（図8ステップ400～403）。

【0075】システムデコーダ10は、入力されたパックをパケットに分割して、パケットデータに含まれるストリームID（stream ID）を参照し、それぞれ指定されたパケットバッファ11-iに格納する。

【0076】このとき最初に現れたPTSデータを図3に示すパケットバッファステータスレジスタ10AのPTS初期値項目に格納する。そして前述した手段で比較して求められたオフセットを図3に示すパケットバッファステータスレジスタ10AのPTSオフセット0に格納する。

【0077】ストリームID0のパケットデータが、パケットバッファ（0）11-0に格納されると、STCモジュール18からのシステムタイムクロックをSTCカウンタ（0）12-0がカウントし始める。

【0078】そしてSTCカウンタ（0）12-0の値からPTSオフセット0を減算した値と、パケットバッファ（0）11-0に格納されているビデオパケットデータのPTS、DTSが同値になると、システムデコーダ10は、ストリーム0のストリーム選択信号（0）を信号線108を介しビデオデコーダ13に通知して、パケットバッファ（0）11-0に格納されているパケットデータから、ビデオストリームを取り出し、システム内部の専用バス102を介してビデオデコーダ13に転送する。

【0079】尚、オーディオパケットであれば、パケットに含まれるオーディオストリームをシステム内部の専用バス102を介してオーディオバッファ16に転送する。オーディオデコーダ17は、オーディオバッファ16に転送されたオーディオデータストリームをデコードし、そのデコードしたアナログオーディオ信号をアナログ信号線107に出力する。

【0080】ビデオデコーダ13では、入力されたストリーム選択信号に従い、レコード情報レジスタ14-0, 14-1, 14-2, 14-3から1つのレジスタを選択し、デコード処理に使用するデコード情報を選択して、当該デコード情報によりデコード処理を行なう。

【0081】デコードされたフレームデータはシステム内部の専用バス105上に出力され、同時にフレーム識別信号が信号線109に出力される。このフレーム識別信号で指定される番号と、デコード情報を選択するときに用いられる番号は同じである。例えば、レコード情報レジスタ14-2のデコード情報2を使用してデコードされたフレームデータは、フレーム識別信号として「2」を指定する。

【0082】上記した動作は1つのシステムストリームをデコードする場合を考えたが、ここでは2つ目のシステムストリームを登録する場合を考える。システムデコーダ10に対して、ID登録レジスタ情報により、配列番号1に登録するビデオパケットのストリームID1を登録する。

【0083】登録の流れは上述した場合と同じであるが、PTSオフセットの算出作業が追加される。システムデコーダ10では、ストリームID1で示されるストリームがパケットバッファ1(11-1)に格納されると、最初に現われるPTS値と既に登録されている、PTS初期値とPTSオフセットの加算値との比較を行ない、同値であればPTSオフセット1に「10」を加算していく。全て比較が終了したときPTSオフセット1の内容が決定する。

【0084】ビデオデコーダ13に対してパケットデータに含まれるビデオストリームの転送のタイミングは、パケットバッファ1(11-1)に含まれるPTS値とSTCカウンタ1(12-1)からPTSオフセット1を減算した値が同値になったとき、ビデオデコーダ13に転送される。

【0085】ここで複数のストリームデータをMPEGハードウェアデコーダ25に供給制御する際の処理について説明する。MPEGハードウェアデコーダ25にシステムストリームを供給制御するためのストリームマルチプレクサドライバ(DRV)に対するデコーダアプリケーションの処理の流れを図5に示している。

【0086】ここでは、動作モードの設定であれば、ストリームマルチプレクサドライバ(DRV)に対して、設定対象のファイルに関しての動作モード(Play/Stop/Pause/Step)の通知を行なう(図5ステップ200, 201)。

【0087】又、ストリームファイルのオープン処理であれば、ストリームマルチプレクサドライバ(DRV)の図7に示すストリーム登録配列データ領域(DA)に対して、デコードを行なうストリームファイル名の登録を行なう(図5ステップ202, 203)。

【0088】この際、MPEGハードウェアデコーダ25に4つのストリーム全てのストリームファイル名が登録されているときは登録失敗となり、ハードウェアデコーダ25に代って、再生指示されたストリームに対応するソフトウェアデコーダの起動、及び再生が行なわれる(図5ステップ204, 205)。

【0089】上記したストリームマルチプレクサドライバ(DRV)の処理の流れを図6に示している。ここでは、上記図5に示すデコーダアプリケーションから動作モードの設定通知を受けると、システムデコーダ10に対してデコード制御レジスタ情報をアクセスして、デコード制御情報の書き込みを行なう(図6ステップ300, 306)。

【0090】また、動作モードの設定でないときは、デコーダアプリケーションから通知されたストリームのファイル名を図7に示すストリーム登録配列データ領域(DA)に登録する(図6ステップ301)。

【0091】この際、ストリームのファイル名が登録できないときはデコーダアプリケーションにエラーを通知して終了する(図6ステップ302, 303)。ストリームのファイル名を登録したときは、通知されたストリームのファイルからストリームIDを取り出し、システムデコーダ10に対してID登録レジスタ情報をアクセスすることにより、ストリームID値の登録を行なう(図6ステップ304, 305)。

【0092】上記したストリームマルチプレクサの内部にある、割り込み処理ルーチンの流れを図8に示している。システムデコーダ10からの割り込み要求がCPU20に受け付けられると、図8に示す割り込み処理ルーチンが実行され、図4(a)に示すステータスレジスタ情報をアクセスして、システムデコーダ10に登録されているストリームのステータス入手し、カウント用の変数Iを「0」に設定する(図8ステップ400)。

【0093】次に配列番号Iの FIFO emptyフラグをチェックし、当該フラグが立っていれば(“1”になつていれば)、配列番号Iで指定されるファイルからパックデータを取り出してシステムデコーダ10に転送する(図8ステップ401, 402, 403)。

【0094】次にカウント用の変数Iをインクリメントし、 $I > 3$ のチェックを行ない、その結果が真となることにより、割り込み処理ルーチンを終了する(図8ステップ404, 405)。

【0095】図9にシステムストリームの構造(図(a))とパックデータの構造(図(b))を示している。パックには、システムヘッダ(system header())とパケットデータ(packet())が含まれる。

【0096】図10にシステムヘッダの構造を示している。システムヘッダの中のStream idが、ビデオまたはオーディオのストリームIDを示している。図11にパケットデータの構造を示している。パケットデータの中ではPTS(presentation time stamp), DTS(de coding time stamp)と、ビデオストリームまたはオーディオストリームに該当するパケットデータサイズ(packetdata byte)を定義している。

【0097】上記したように、本発明の実施形態によれば、MPEGデータストリームを扱うシステムデコーダ10に、複数のビデオパケットをそれぞれ個別に貯える複数個のパケットバッファ11-0, 11-1, 11-2, 11-3と、このパケットバッファ11-0, 11-1, 11-2, 11-3に貯えられたデータをSTCモジュール18より供給されるシステムタイムクロックに同期して出力制御する機能とを設け、ビデオデコーダ13に、複数のデータストリームをデコードするための各データストリ

ームに対応する複数組のデコード情報をそれぞれ個別に格納する複数個のレコード情報レジスタ14-0, 14-1, 14-2, 14-3と、このレジスタ14-0, 14-1, 14-2, 14-3に対応して、デコード対象となるデータを貯える複数個のデータバッファ(FIFO)と、上記レジスタの1つを選択し、当該レジスタ14-iに貯えられたデコード情報をもとに對応するデータバッファ(FIFO)に貯えられたデータをデコード処理するデコーダ本体とを設けて、单一のハードウェアデコーダにより複数の動画像データストリームを扱うことができる構成としたことにより、複数のMPEGデータストリームを並行して扱うことのできるMPEGハードウェアデコーダを実現する際に、同時に扱う複数のMPEGデータストリームそれぞれに対して独立したMPEGハードウェアデコーダを設けることなく、簡単な構成で、安価にしかもコンパクトに実現できる。

#### 【0098】

【発明の効果】以上詳記したように本発明によれば、ハードウェアデコーダ内部に複数のMPEGデータストリームを再生可能にするための複数組のバッファ、及び複数のストリーム制御機能を設けたことにより、单一のハードウェアリソースで複数の動画像データストリームをリアルタイムに再生できる動画像デコード装置が提供できる。

【0099】即ち本発明によれば、動画像データストリームを扱うシステムデコーダに、複数のビデオパケットをそれぞれ個別に貯える複数個のパケットバッファと、この複数個のパケットバッファに貯えられたデータを外部より供給される信号に同期して出力制御する手段と、上記複数個のパケットバッファの状態を外部に通知する手段とを具備してなる構成としたことにより、单一のシステムデコーダで複数の動画像データストリームをリアルタイムに再生することができる。

【0100】又、本発明によれば、動画像データストリームを扱うビデオデコーダに、複数の動画像データストリームをデコードするための各データストリームに対応する複数組のデコード情報をそれぞれ個別に格納する複数のレジスタを内蔵してなる構成としたことにより、複数の動画像データストリームの再生を单一のビデオデコーダで実行できる。

【0101】又、本発明によれば、動画像データストリームを扱うビデオデコーダに、複数の動画像データストリームをデコードするための各データストリームに対応する複数組のデコード情報をそれぞれ個別に格納する複数個のレジスタと、このレジスタに対応して設けられ、それぞれデコード対象となるデータを貯える複数個のFIFOバッファと、上記レジスタの1つを選択し、当該レジスタに貯えられたデコード情報をもとに對応する FIFOバッファに貯えられたデータをデコード処理するデコーダ本体とを具備してなる構成としたことにより、

複数の動画像データストリームの再生を单一のビデオデコーダで実行できる。

【0102】又、本発明によれば、動画像データストリームを再生する動画像デコード装置に於いて、複数の動画像データストリームをそれぞれ個別に貯える複数の記憶手段をもつシステムデコーダと、このシステムデコーダの記憶手段に貯えられた複数の動画像データストリームを再生するための複数組のデコード情報をそれぞれ個別に貯える複数の記憶手段をもつビデオデコーダとを具備してなる構成したことにより、单一のハードウェアリソースで複数の動画像データストリームをリアルタイムに再生できる。

【0103】又、本発明によれば、サウンドデータを含む複数の動画像データストリームをそれぞれ個別に貯える複数の記憶手段をもつシステムデコーダと、このシステムデコーダの記憶手段に貯えられた複数の動画像データストリームを再生するための複数組のデコード情報をそれぞれ個別に貯える複数の記憶手段をもつビデオデコーダと、このビデオデコーダのフレームデータ処理を含むデコード処理に供される作業用バッファと、上記データストリーム中のサウンドデータを動画像に同期して再生出力するサウンドバッファ及びサウンドデコーダとを備えてMPEGデコーダを構成したことにより、单一のハードウェアリソースで複数の動画像データストリームをリアルタイムに再生できる。

【図面の簡単な説明】  
【図1】本発明の実施形態によるMPEGハードウェアデコーダの構成を示すブロック図。

【図2】図1に示すMPEGハードウェアデコーダを含む、MPEGデコードシステムの構成を示すブロック図。

【図3】上記実施形態に於けるMPEGハードウェアデコーダ25のシステムデコーダ10に設けられたパケットバッファステータスレジスタ10Aの内部レジスタ配列構成を示す図。

【図4】上記実施形態に於ける、システムデコーダ10内の制御レジスタ10B上に生成される各制御情報のデータ構造を示す図。  
【図5】上記実施形態に於ける、RAM22上に置かれたストリームマルチプレクサドライバ(DRV)を制御するためのデコーダアプリケーションでのストリームマルチプレクサ処理の制御手順を示すフローチャート。

【図6】上記実施形態に於ける、RAM22上に置かれたストリームマルチプレクサドライバ(DRV)により実行されるストリームマルチプレクサ処理の手順を示すフローチャート。  
【図7】上記実施形態に於ける、RAM22上に置かれたストリームマルチプレクサドライバ(DRV)のストリーム登録配列データ領域(DA)の構成を示す図。

【図8】上記実施形態に於ける、ストリームマルチプレ

17

クサ処理に於ける割り込み処理ルーチンの流れを示すフローチャート。

【図9】システムストリームの構造(a)と、パックデータの構造(b)を示す図。

【図10】システムヘッダの構造を示す図。

【図11】パケットデータの構造を示す図。

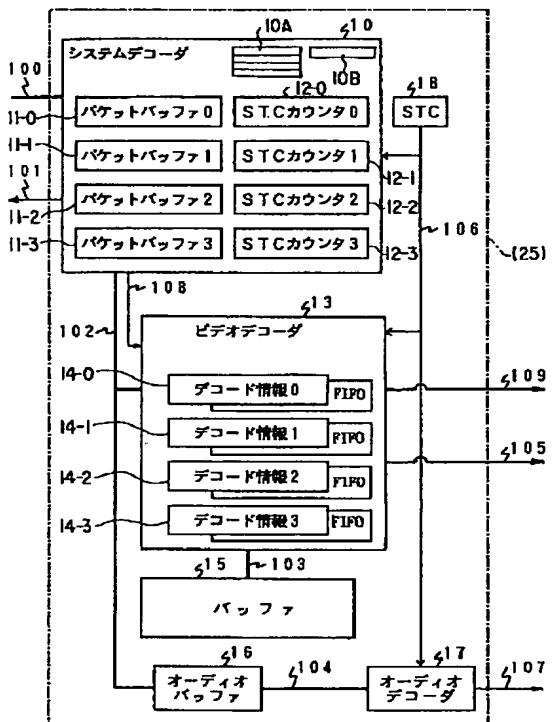
【符号の説明】

10…システムデコーダ、10A…パケットバッファスデータレジスタ、10B…制御レジスタ、11-0, 11-1, 11-2, 11-3…パケットバッファ、12-0, 12-1, 12-2, 12-3…パケットバッファ、13…ビデオ

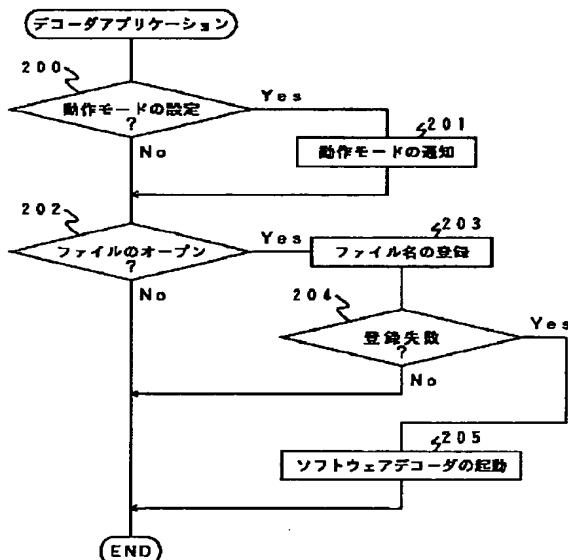
10

デコーダ、14-0, 14-1, 14-2, 14-3…レコード情報レジスタ、15…デコード用バッファ、16…オーディオバッファ、17…オーディオデコーダ、18…STC(システムタイムクロック)モジュール、20…CPU、21…割り込みコントローラ、22…RAM、23…SCSIインターフェース、24…CDドライブ、25…MPEGハードウェアデコーダ、26…オーディオ増幅器(AMP)、27…スピーカ、28…表示コントローラ(VGA)、29…モニタ、DRV…ストリームマルチプレクサドライバ、DA…ストリーム登録配列データ領域。

【図1】



【図5】

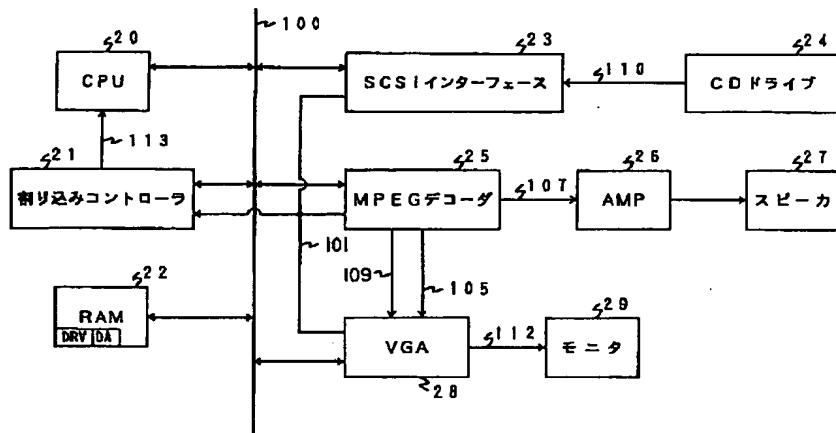


【図3】

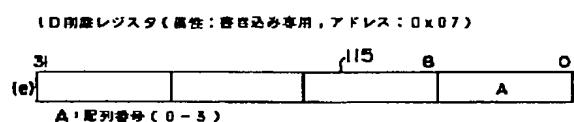
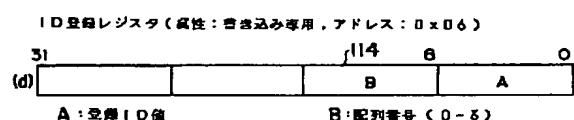
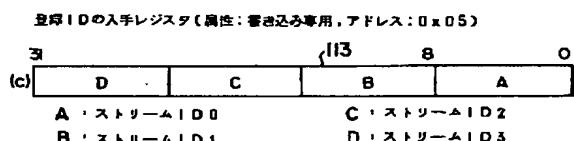
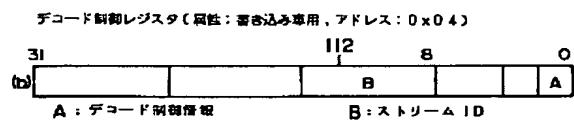
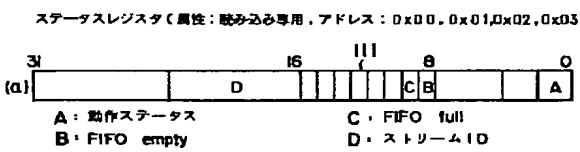
配列番号	ストリームID	動作モード	PTSオフセット	PTS初期値
0	ストリームID0	動作モード0	PTSオフセット0	PTS初期値0
1	ストリームID1	動作モード1	PTSオフセット1	PTS初期値1
2	ストリームID2	動作モード2	PTSオフセット2	PTS初期値2
3	ストリームID3	動作モード3	PTSオフセット3	PTS初期値3

10A

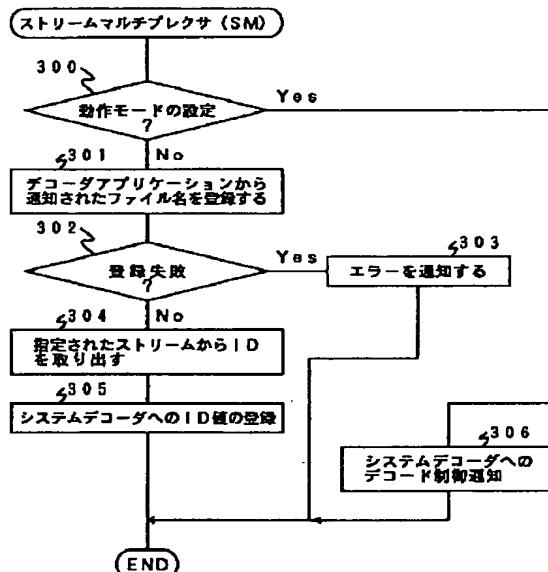
【図2】



【図4】



【図6】



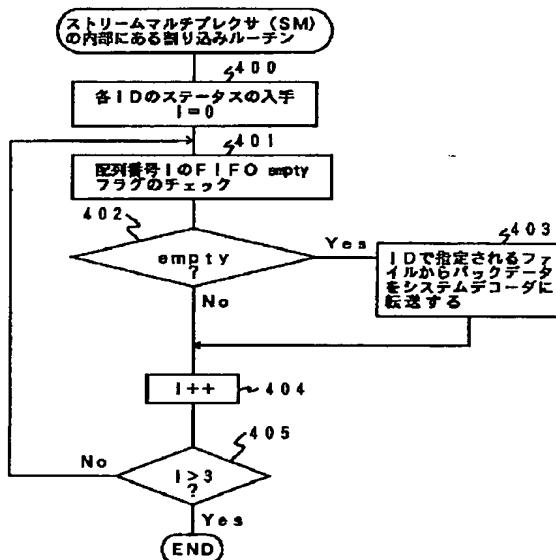
【四七】

記別番号	ファイル名	ストリーム ID	動作モード
0	ファイル名 0	ストリーム ID 0	動作モード 0
1	ファイル名 1	ストリーム ID 1	動作モード 1
2	ファイル名 2	ストリーム ID 2	動作モード 2
3	ファイル名 3	ストリーム ID 3	動作モード 3

1  
DA

【 9】

(a)



〔四一〇〕

```
    } while(nextbits( )>=pack_start_code)
    iso_11172_end_code
}

```

Syntax	No. of bits	Mnemonic
pack( ) {		
pack_start_code 0X000001BA	32	bs1bf
'010'	4	bs1bf
system_clock_reference [32..30]	3	bs1bf
marker_bit	1	bs1bf
system_clock_reference [29..15]	15	bs1bf
marker_bit	1	bs1bf
system_clock_reference [14..0]	15	bs1bf
marker_bit	1	bs1bf
marker_bit	1	bs1bf
aux_rate	22	uimsbf
marker_bit	1	bs1bf
 If(nextbits( ) == system_header_start_code)		
system_header( )		
 while(nextbits( ) == packet_start_code_prefix)		
packet( )		

Syntax		Nb. of bits	Mnemonic
system_header( ) {			
system_header_start_code	0X000001BB	32	bslbf
header_length		16	uiabsbf
marker_bit		1	bslbf
rate_bound		22	uiabsbf
marker_bit		1	bslbf
audio_bound		6	uiabsbf
fixed_flag		1	bslbf
C P S_flag		1	bslbf
system_audio_lock_flag		1	bslbf
system_video_lock_flag		1	bslbf
marker_bit		1	bslbf
video_bound		5	uiabsbf
reserved_byte		8	uiabsbf
while(nextbit( ) == 1) {			
stream_id		8	uiabsbf
'1'		2	bslbf
STD_buffer_bound_scale		1	bslbf
STD_buffer_size_bound		13	uiabsbf

【図 11】

Syntax	No. of bits	Mnemonic
packet( ) {		
packet_start_code_prefix 0X000001	24	bslbf
stream_id	8	uimsbf
packet_length	16	uimsbf
if(stream_id!=private_stream_2) {		
while(nextbits( )>='1111 1111')		
stuffing_byte	8	bslbf
if(nextbits( )=='01') {		
'01'	2	bslbf
S_TD_buffer_scale	1	bslbf
S_TD_buffer_size	13	uimsbf
}		
if(nextbits( )=='0010') {		
'0010'	4	bslbf
presentation_time_stamp[32..30]	3	bslbf
marker_bit	1	bslbf
presentation_time_stamp[29..15]	15	bslbf
marker_bit	1	bslbf
presentation_time_stamp[14..0]	15	bslbf
marker_bit	1	bslbf
}		
else if(nextbits( )=='0011') {		
'0011'	4	bslbf
presentation_time_stamp[32..30]	3	bslbf
marker_bit	1	bslbf
presentation_time_stamp[29..15]	15	bslbf
marker_bit	1	bslbf
presentation_time_stamp[14..0]	15	bslbf
marker_bit	1	bslbf
'0001'	4	bslbf
decoding_time_stamp[32..30]	3	bslbf
marker_bit	1	bslbf
decoding_time_stamp[29..15]	15	bslbf
marker_bit	1	bslbf
decoding_time_stamp[14..0]	15	bslbf
marker_bit	1	bslbf
}		
else	'0000 1111'	bslbf
}		
for(i=0;i<4;i++) {		
packet_data_byte	8	bslbf
}		